Docket No.: 95-307 PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of :
:
KANURI :
:
Serial No.: 09/519,848 : Group Art Unit: 2663
:
Filed: March 6, 2000 : Examiner: GEORGE, Keith M.

For: SELECTIVE ADDRESS TABLE AGING IN A NETWORK SWITCH BASED ON APPLICATION STATE DETERMINED FROM A RECEIVED DATA PACKET

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Sir:

This is an appeal from the final rejection of claims 1-2 and 8-11 in the above-identified patent application.

This Appeal Brief is submitted in triplicate as required by 37 C.F.R. §1.192(a).

1. Real Party in Interest:

This application is assigned to Advanced Micro Devices, Inc., the real party of interest.

2. Related Appeals and Interferences:

There are no other appeals or interferences known to Appellant that will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

3. Status of Claims:

Claims 1-13 are pending in this application. Claims 1-2 and 8-11 stand rejected by the Examiner, and dependent claims 3-7 and 12-13 are indicated by the Examiner as reciting allowable subject matter. Claims 1-2 and 8-11 are appealed.

4. Status of any Amendment File Subsequent to Final Rejection:

No Amendment was filed in response to the Final Rejection. A Response to the Final Rejection was filed on March 15, 2004.

5. Summary of Invention:

The invention relates to a network switch (e.g., 12a of Figure 1), configured for performing layer 2 (and above) switching in a local area network (10 of Figure 1), such as an Ethernet (IEEE 802.3) network, without blocking of incoming data packets (see, e.g., page 4, lines 16-23 of the specification, and Abstract). The network switch (12) includes a switch fabric (25 of Figure 1) configured for executing layer 2 switching decisions for received data packets, for example based on source address, destination address, and VLAN information within a layer 2 header (see, e.g., page 4, lines 27-30 of the specification). The switch fabric 25 executes its layer 2 switching decisions based on accessing a network switch address table (82 of Figure 4) having address entries (84 of Figure 4) (see, e.g., page 8, lines 12-23 of the specification).

The invention addresses the problem of deleting aged address entries from the network switch address table to prevent overflowing of the network switch address table, while avoiding premature deletion of an address entry that would require the network switch to relearn the

network addresses (see page 1, line 20 to page 2, line 18 of specification). The invention selectively deletes (step 102 of Figure 5) address entries (84) based on determining from the payload of a received packet (step 98 of Figure 5) an application state (e.g., step 100 of Figure 5) of a prescribed network application (e.g., HTTP, SNMP, FTP, Telnet, etc.) being executed between two network nodes (see, e.g., page 5, lines 15-22 and page 9, lines 13-20 of the specification). Hence, the deletion of address entries can be precisely controlled based on the completion of a network application session, as determined by the application state from the received layer 2 data packet (see e.g., page 3, lines 3-5 and 17-20 and page 9, lines 26-29 of the specification).

Hence, the invention of claim 1 is directed to a method in an integrated network switch. The method includes determining an application state (step 98) for a prescribed network application from a received layer 2 data packet. The method also includes selectively deleting (steps 100, 102 of Figure 5) an address entry (84 of Figure 4) from a network switch address table (82 of Figure 4) that specifies at least one of a source of the received layer 2 data packet and a destination of the layer 2 data packet, based on the determined application state (step 100 of Figure 5).

The invention of claim 2 adds to the invention of claim 1 by storing within a network switch port (20 of Figure 1), having received the received layer 2 data packet, a plurality of templates (e.g., 62a, 62b, 62c, and 62d of Figures 3A or 3B) configured for identifying the application state from respective available application states of the prescribed network application (page 4, lines 11-13; page 5, lines 29-33; page 7, line 34 to page 8, line 11).

The invention of claims 3-7 are not involved in the appeal and therefore will not be

discussed herein.

The invention of claim 8 adds to the invention of claim 1 by: (1) initiating an application-specific aging timer (84b of Figure 4) configured for counting an application-specific aging interval for the address entry in response to determining the application state (step 96 of Figure 5, page 8, lines 20-23; page 9, lines 7-10); and (2) deleting the address entry if the address entry is not accessed upon expiration of the application-specific aging interval (steps 106 and 102 of Figure 5, page 9, lines 7-13 and 21-25).

The invention of claim 9 adds to the invention of claim 8 by resetting (step 104 of Figure 5) the application-specific aging timer in response to detecting (step 98 of Figure 5) an access of the address entry during the application-specific aging interval (page 9, lines 17-20).

The invention of claim 10 provides a network switch comprising network switch ports (20 of Figure 1), and switching logic (25 of Figure 1). Each switch port includes a packet classifier (24 of Figures 1 and 2) configured for determining an application state for a detected one of a plurality of prescribed network applications from a received layer 2 data packet (step 92 of Figure 5, page 8, line 29 to page 9, line 6). The switching logic is configured for selectively deleting (step 102 of Figure 5) an address entry (84 of Figure 4) that specifies at least one of a source of the received layer 2 data packet and a destination of the layer 2 data packet (84c of Figure 4), based on one of the determined application state (step 100 of Figure 5) and a determined inactivity of the address entry during an application-specific aging interval (step 106 of Figure 5) based on the detected one prescribed network application (step 96 of Figure 5; page 8. lines 7-25).

The invention of claim 11 is that the switching logic of claim 10 includes a

programmable timer (e.g., T1 of Figure 4) configured for initiating counting of the application-specific aging interval for the address entry (e.g., "Start1" of Figure 4) in response to detection of the application state from the received layer 2 data packet.

The invention of claims 12 and 13 are not under appeal and therefore will not be discussed herein.

6. Issues:

A. Whether claims 1 and 10 are patentable under 35 U.S.C. §102(e) as not having been anticipated by U.S. Patent No. 6,094,435 to Hoffman et al. (hereinafter "Hoffman").

B. Whether claim 2 is patentable under 35 U.S.C. §103(a) as not having been obvious in view of Hoffman and U.S. Patent No. 5,128,926 to Perlman et al. (hereinafter "Perlman").

C. Whether claims 8-9 and 11 are patentable under 35 U.S.C. §103(a) as not having been obvious in view of Hoffman and U.S. Patent No. 6,104,696 to Kadambi et al. (hereinafter "Kadambi").

7. Grouping of Claims:

With regard to the rejections, claims 1 and 10 stand separately and do not stand or fall together; claims 8-9 stand or fall together, but claim 11 stands separately from claims 8-9 and does not stand or fall together with claims 8-9.

8. Arguments:

A1. **Claim 1 is patentable under 35 U.S.C. §102(e) over Hoffman.**

In the Final Office Action, the Examiner rejected claim 1 under 35 USC §102(e) in view of Hoffman. Claim 1 is neither anticipated nor rendered obvious by Hoffman for the following reasons.

Claim 1 specifies a method in an integrated network switch, including "determining an application state for a prescribed network application from a received layer 2 data packet." As described in the specification, the term "network application" refers to higher-protocol communications between two network nodes that involve multiple "application states": network nodes communicate according to a prescribed network application (e.g., OSI Layer 7 ("Application Layer") protocols such as HTTP, SNMP, FTP, Telnet, etc.), resulting in prescribed data flows between the two network nodes; hence, the layer 2 data packets transferred between the network nodes will include payload information that specifies the prescribed network application state, for example a request to initiate a session, acknowledgment, communication during the session, a request to terminate the session, and acknowledgment of termination of the session (see, e.g., page 5, lines 15-22 and page 8, lines 2-7).

Hence, the claimed "determining an application state for a prescribed network application from a received layer 2 data packet" enables the integrated network switch to determine the state of the network application between two network nodes (see, e.g., page 8, lines 7-11).

Claim 1 also specifies "selectively deleting an address entry from a network switch address table ... based on the determined application state." Hence, the deletion of an address entry can be precisely controlled based on the completion of a network application session between two network nodes, as determined by the application state from the received layer 2 data packet (see e.g., page 3, lines 3-5 and 17-20 and page 9, lines 26-29 of the specification).

Hoffman discloses a switching element 36 that performs forwarding functions using layer 2 and layer 3 information, and possibly also some layer 4 information (col. 8, lines 56-60). Hoffman adopts the OSI model (col. 1, lines 55-58) that models network operations using seven (7) layers, and Hoffman further specifies that "the relevant layers for describing this invention include OSI Layers 1 through 4." (Col. 2, lines 5-6). Hoffman further acknowleges that the OSI Layers 1 through 4 correspond to the physical layer, the data link layer, the network layer, and the transport layer, respectively.

Hoffman further recognizes that Layer 4, the transport layer, is intended to provide a port address for network applications:

> A key difference between the transport layer and the lower layers is that an application on a source end station can carry out a conversation with a similar application on a destination end station anywhere in the network; whereas the lower layers [Layers 1 through 3] carry on conversations with end stations which are its immediate neighbors in the network.

(Col. 2, lines 31-31-36).

Hence, both Hoffman and the claims on appeal are consistent in their terminology, namely describing "network applications" as applications executed on respective end stations that communicate by relying on services provided by the lower OSI layers. A copy of the Exhibit B attached to the Response Filed October 14, 2003 is attached for convenience to further demonstrate the interpretation applied in the art to the OSI layers 1-7.

The Final Action asserts that "Destination aging in the network element indicates which layer 2 and layer 3 entries are active (determine an application state). The information implements in accordance with IEEE standard 802.1d type address aging (delete an address entry from a network switch address table based on the application stage) (column 16, lines 43-53)."

Hoffman does not disclose or suggest determining an application state for a prescribed network application from a received layer 2 data packet, as claimed. Rather, the cited portion (col. 16, lines 43-53) indicates IEEE standard 802.1d type address aging is to be used, where " the information for an entry is updated <u>every time an entry is matched</u>, either by a layer 2 destination search or a layer 3 match cycle for the entry" (col. 16, lines 51-53). Hence, Hoffman explicitly teaches that aging is based on identifying whether a <u>matching address</u> is found within a specified aging interval, in accordance with IEEE 802.1D.

Attached for convenience is a copy of the Exhibit A attached to the Response Filed October 14, 2003. Exhibit A showed on page 2 that the IEEE 802.1D Specification (1998) specifies on page 45 that entries "shall be automatically removed after a specified time, the Ageing [sic] Time, has elapsed since the entry was created or last updated." Further , Table 7-4 of the IEEE 802.1D Specification specifies a range of applicable aging parameter values with <u>a recommended default value</u>.

Hence, Hoffman merely discloses that aging is based on whether a <u>matching address</u> (layer 2 or layer 3) is detected before expiration of the aging interval. As described in the Background Art of the subject application, however, removing address entries based on a specified time may cause either premature deletion of an address entry (requiring relearning), or may risk overflowing the network switch address table.

Hence, contrary to the assertions in the Final Action, Hoffman does not disclose or suggest determining an application state for a prescribed network application from a received layer 2 data packet, let alone selectively deleting an address entry based on the determined application state. Rather, Hoffman merely discloses that an address entry is deleted if a matching

address is not detected before expiration of a <u>fixed</u> aging interval.

The Final Action asserts on page 4 (paragraph 10) that "Determining which layer 2 and layer 3 entries are active is equivalent to determining an application state for a prescribed network application." This assertion is both factually and legally improper.

As shown above, both the subject application and Hoffman distinguish between *link activity for layer 2 and layer 3 address entries* (see col. 2, lines 34-36 "the lower layers [1-3] carry on conversations with end stations which are its immediate neighbors in the network"; and col. 16, lines 49-51 "[d]estination aging in the network element 12 indicates which layer 2 and layer 3 entries are active") and *application operations between network nodes* (see col. 2, lines 32 "an application on a source end station can carry out a conversation with a similar application on a destination end station anywhere in the network"). Further, Hoffman explicitly specifies that "the relevant layers for describing this invention include OSI Layers 1 through 4" (col. 2, lines 5-6). There is no reference whatsoever in Hoffman that the disclosed IEEE 802.1d type address aging should be interpreted as anything other than conventional Layer 2 type aging.

In fact, Hoffman discloses that <u>only the header of the packet</u> (i.e. layer 2, layer 3, possibly layer 4 headers) is sent to the forwarding logic 52 (Fig. 3, col. 9, lines 19-26). Hence, Hoffman is <u>unable</u> to determine the application state because the relevant information is never presented to the class logic 60 in the forwarding logic 52 (Fig. 4, col. 12, lines 30-38).

The assertion of equivalency further is legally improper as applying an unreasonable interpretation to the claims. The claimed "applicaton state for a prescribed network application" cannot be interpreted so broadly as to encompass layer 2 or layer 3 link state. Hence, "claims are not to be read in a vacuum, and limitations therein are to be interpreted in light of the

specification in giving them their 'broadest reasonable interpretation.'" MPEP § 2111.01 at 2100-37 (Rev. 1, Feb. 2000) (quoting In re Marosi, 218 USPQ 289, 292 (Fed. Cir. 1983)(emphasis in original)).

Moreover, the assertion of equivalency is without legal foundation: "A prior art patent is a reference only for that which it teaches." Corning Glass v. Sumitomo Electric, 9 USPQ2d 1962, 1970 (Fed. Cir. 1989). Equivalency is not proper in a §102 rejection. The burden is on the Examiner to demonstrate that Hoffman discloses each and every element of the claim. See MPEP 2131. "The identical invention must be shown in as complete detail as is contained in the ... claim." Richardson v. Suzuki Motor Co., 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). "Anticipation cannot be predicated on teachings in the reference which are vague or based on conjecture." Studiengesellschaft Kohle mbH v. Dart Industries, Inc., 549 F. Supp. 716, 216 USPQ 381 (D. Del. 1982), aff'd., 726 F.2d 724, 220 USPQ 841 (Fed. Cir. 1984).

Further, the Advisory Action demonstrates the flawed reasoning of the rejection: its assertion of inherency ("[a]pplication data is inhereint [sic] to the payload of a layer 2 packet") is coupled with the assertion that "[o]ne of ordinary skill in the art would understand that if there is no data at the application layer then there will not be a layer 2 packet and the state will appear as inactive." (Emphasis Added).

Although the Advisory Action raises an obviousness argument (tacitly admitting the deficiency of the rejection), the obviousness argument is telling in its rationale that the state will appear as inactive because "there will not be a layer 2 packet." In other words, the Advisory Action asserts that the inactive state is determined based on the absence of a layer 2 packet. However, claim 1 specifies "determining an application state for a prescribed network from a

received layer 2 data packet." As quoted above, anticipation cannot be predicated on teachings in the reference which are vague or based on conjecture.

For these and other reasons, the §102 rejection of claim 1 should be reversed.

A2. **Claim 10 is further patentable under 35 U.S.C. §102(e) over Hoffman.**

Claim 10 specifies "determining an application state ... from a received layer 2 data packet" and "selectively deleting an address entry ... based on ... the determined applicatoin state...." Hence, the arguments above with respect to claim 1 are relevant and are incorporated in their entirety herein by reference.

Claim 10 further specifies a network switch having a plurality of network switch ports, "each including a packet classifier configured for determining an application state for a detected one of a plurality of prescribed network applications from a received layer 2 data packet."

Hence, the packet classifier (e.g., 24 of Figures 1 and 2) enables each corresponing network switch port (e.g., 20 of Figures 1 and 2) to identify whether the received layer 2 data packet is carrying data for a prescribed network application (e.g., HTTP, SNMP, Telnet), and the application state of that prescribed network application (see, e.g., page 7, line 32 to page 8, line 11), providing distributed processing that reduces processing requirements for the switch fabric 25.

The Final Action fails to address this claimed feature, but rather relies on the assertions as identified above with respect to claim 1. Nevertheless, Hoffman neither discloses nor suggests a plurality of network switch ports, where each network switch port includes a packet classifier for determining an application state, as claimed.

Hoffman discloses in Fig. 3 a switching element 36 having a plurality of input ports 50 (e.g., 50a, 50b, ... 50n), a plurality of output ports 56, and forwarding logic 52. Each input port 50 receives a packet, tests the packet for correctness (i.e., verifying it is not ill-formed), then forwards the header to the forwarding logic 52 (col. 9, lines 19-26). The forwarding logic 52 includes class logic 60 (Figure 4) that determines "any encapsulation information and to determine a class for the layer 3 information." (See col. 12, lines 29-37).

Hence, Hoffman discloses that the class logic 60 is positioned centrally within the switching element 36. Hence, Hoffman neither discloses nor suggests implementing the class logic 60 within each network switch port, as claimed. Regardless, there is no discloure or suggestion in Hoffman to determine an application state for a detected one of a plurality of prescribed network applications from a received layer 2 data packet, as claimed.

For these and other reasons, the §102 rejection of claim 10 should be reversed.

B. **Claim 2 is patentable under 35 U.S.C. §103(a) over Hoffman and Perlman.**

Claim 2 depends from claim 1, hence the arguments above with respect to claim 1 are incorporated in their entirety herein by reference.

Claim 2 specifies storing <u>within a network switch port</u> having received the received layer 2 data packet a plurality of templates configured for identifying the application state from respective available application states of the prescribed network application (page 7, line 34 to page 8, line 11).

Hence, the templates enable the network switch port <u>having received the received layer 2 data packet</u> to identify the determined application state from respective available application

states. Consequently, the storage of the templates within the network switch port enables the network switch port to determine the application state and forward the information to the a centralized switch fabric, reducing processing requirements by the centralized switching fabric (see page 9, lines 11-20).

The Final Action does not address this claimed feature, but rather asserts that in paragraph 4 that:

> Perlman teaches that information is communicated through a network along a myriad of links interconnected by nodes. Each node must know the states (e.g., operative or inoperative)(identify the application state) of the links in the network in order to send packets along effective paths [citation omitted]... [I]t would have been obvious ... to use the knowledge of application state taught by Perlman in the aging method of Hoffman.

As argued above with respect to claim 1, the rejection mischaracterizes the teachings of Perlman by suggesting that "link state" teaches the claimed "application state". The "link state packet" (LSP) of Perlman refers to a link state routing protocol, where each router provides link status for <u>connected links</u> to other routers in the network. Regardless, the rejection fails to address the claimed feature of storing <u>within the network switch port</u> a <u>plurality of templates</u> configured for identifying the application state.

Hence, the hypothetical combinaton of Hoffman and Perlman does not address the claimed features of storing a plurality of templates <u>in a network switch port</u>, let alone the problems addressed by the claimed features. An evaluation of obviousness must be undertaken from the perspective of one of ordinary skill in the art addressing the same problems addressed by the applicant in arriving at the claimed invention. <u>Bausch & Lomb, Inc. v. Barnes-Hind/Hydrocurve</u>, 23 USPQ 416, 420 (Fed. Cir. 1986), <u>cert. denied</u>, 484 US 823 (1987). Thus,

the claimed structures and methods cannot be divorced from the problems addressed by the inventor and the benefits resulting from the claimed invention. In re Newell, 13 USPQ2d 1248, 1250 (Fed. Cir. 1989).

Further, the Final Action at page 5, paragraph 11, makes an improper reference to inherency in an attempt to create the presence of a "template":

> In order for Perlman to determine the operative or inoperative status of the links, it would be inherent [sic] for Perlman to have a reference to use to determine the status; this reference in each node is a template.

(Page 5, paragraph 11, lines 8-10 of Final Office Action).

This assertion improperly relies on inherency: Inherency is not applicable in a rejection under §103. In re Newell, 13 USPQ2d 1248, 1250 (Fed. Cir. 1989). Regardless, the above assertions in the Final Action still do not address the feature of storing the plurality of templates in the network switch port.

For these and other reasons, the Final Action fails to establish a prima facie case of obviousness; hence, the §103 rejection of claim 2 should be reversed.

C1. Claims 8-9 are patentable under 35 U.S.C. §103(a) over Hoffman and Kadambi.

Claim 8 depends from claim 1 and claim 9 depends from claim 8. Hence, the arguments above with respect to claim 1 are incorporated in their entirety herein by reference.

Claim 8 further specifies initiating an application-specific aging timer (e.g., 84b of Figure

4) configured for counting an application-specific aging interval for the address entry in response to the application state (see, e.g., step 96 of Figure 5, page 8, lines 20-23; page 9, lines 7-10). Hence, claim 8 specifies the additional feature that an application-specific aging interval may be used, based on determining the application state for the prescribed network application. Hence, different application-specific aging timers may be used for respective network applications (e.g., HTTP, SNMP, FTP, Telnet, etc.).

The Final Action asserts that Kadambi teaches that each Ethernet Port Interface Controller (EPIC) is provided with an age timer (Fig. 18, step 18-1), and that if the hit bit is not set upon expiration of the age timer, the ARL entry is deleted, citing col. 22, line 63 to col. 23, line 17.

Again the Final Action mischaracterizes the applied reference and the claimed "determining the application state" by improperly equating link activity on a connected link with application state between two network applications executed on remote network nodes. As described on page 1, line 20 to page 2, line 5 of the specification, the use of a "hit bit" described in Kadambi is a conventional layer 2 operation for resetting the <u>fixed</u> aging timer specified by the IEEE 802.1D Specification (See attached Exhibit A from Response Filed Oct. 14, 2003).

Although Kadambi <u>does</u> teach that the aging process is performed only on entries that belong to the module having received the data packet, this teaching has no relation to the assertion in the Final Action that "the aging timer is specific to the application described". Rather, as described at col. 22, lines 31-36 and 43-47 the aging timer is specific to the <u>port having received the data packet</u>.

Further, column 22, lines 64-65 specify that "an age timer is provided within each EPIC

module 20 and GPIC module 30". Hence, col. 22, line 63 to col. 23, line 27 are directed to an aging timer used by an aging process to delete aged entries that are stored in the owner of the address. Hence, the age timers of Kadambi are port-specific aging timers based on the port having received the packet carrying the learned address.

Neither Hoffman or Kadambi, singly or in combination, disclose nor suggest an application-specific aging timer configured for counting the application-specific aging interval for a prescribed network application, as claimed. For these and other reasons, claims 8-9 are patentable over Hoffman and Kadambi. Hence, the Final Action fails to establish a prima facie case of obviousness; accordingly, this §103 rejection of claims 8-9 should be reversed.

C2. **Claim 11 is are patentable under 35 U.S.C. §103(a) over Hoffman and Kadambi.**

Claim 11 depends from claim 10, hence, the arguments above with respect to claim 1 are incorporated in their entirety herein by reference.

Further, claim 11 specifies that the switching logic includes a programmable timer configured for initiating counting (step 96) of the application-specific aging interval for the address entry in response to detection of the application state from the received layer 2 data packet. Hence, the arguments above with respect to claims 8-9 are incorporated in their entirety herein by reference.

As described above with respect to claim 10, the Final Action does not address that each network switch port includes a packet classifier configured for determining an application state for a detected one of a plurality of prescribed network applications from a received layer 2 data

packet. Further, the Final Action fails to address that the programmable timer in the switching logic is configured for initiating counting of the application-specific aging interval in response to detection of the application state by the packet classifier in the network switch port having received the packet.

Hence, the Final Action fails to establish a prima facie case of obviousness for claim 11. For these and other reasons, the §103 rejection of claim 11 should be reversed.

Conclusion

For the reasons set forth above, it is clear that Appellant's claims 1-2 and 8-11 are patentable over the references applied. Accordingly the appealed claims 1-2 and 8-11 should be deemed patentable over the applied references. It is respectfully requested that this appeal be granted and that the Examiner's rejections be reversed.

To the extent necessary, Appellant petitions for an extension of time under 37 C.F.R. 1.136. Please charge any shortage in fees due in connection with the filing of this paper, including any missing or insufficient fees under 37 C.F.R. 1.17(a), to Deposit Account No. 50-0687, under Order No. 95-307, and please credit any excess fees to such deposit account.

Respectfully submitted,

Manelli Denison & Selter, PLLC

Leon R. Turkevich
Registration No. 34,035

Customer No. 20736
Attached: Exhibits A, B from Response filed Oct. 14, 2003

APPENDIX – CLAIMS ON APPEAL

1. (ORIGINAL) A method in an integrated network switch, the method comprising:

determining an application state for a prescribed network application from a received layer 2 data packet; and

selectively deleting an address entry from a network switch address table that specifies at least one of a source of the received layer 2 data packet and a destination of the layer 2 data packet, based on the determined application state.

2. (ORIGINAL) The method of claim 1, further comprising storing within a network switch port having received the received layer 2 data packet a plurality of templates configured for identifying the application state from respective available application states of the prescribed network application.

8. (ORIGINAL) The method of claim 1, further comprising:

initiating an application-specific aging timer configured for counting an application-specific aging interval for the address entry in response to determining the application state; and

deleting the address entry if the address entry is not accessed upon expiration of the application-specific aging interval.

9. (ORIGINAL) The method of claim 8, further comprising resetting the application-specific aging timer in response to detecting and access of the address entry during the application-specific aging interval.

10. (PREVIOUSLY PRESENTED) A network switch, comprising:

network switch ports, each including a packet classifier configured for determining an application state for a detected one of a plurality of prescribed network applications from a received layer 2 data packet; and

switching logic configured for selectively deleting an address entry that specifies at least one of a source of the received layer 2 data packet and a destination of the layer 2 data packet, based on one of the determined application state and a determined inactivity of the address entry during an application-specific aging interval based on the detected one prescribed network application.

11. (ORIGINAL) The switch of claim 10, wherein the switching logic includes a programmable timer configured for initiating counting of the application-specific aging interval for the address entry in response to detection of the application state from the received layer 2 data packet.